



Figure 1

A conceptual distribution of code portability shows that most of your

software may be portable code that relies on a compiler to abstract the porting effort. A smaller portion of your software may rely on support files, such as an operating system, common libraries, and drivers. Architecturally optimized code extracts the greatest performance from the old environment and adds the most complexity to your porting effort.